조사 과제#3

1. NPU 의 기본 개념 및 NPU 구조

NPU 의 기본 개념

NPU(Neural Processing Unit) 딥러닝 네트워크를 구성하는 각 레이어를 실리콘으로 구현한 칩셋이다. 주로 인공지능(AI) 연산을 위해 설계된 특수 프로세서로, 주로 딥러닝 작업에 최적화되어 있으며, 뉴럴 네트워크 연산을 효율적으로 수행하는데 중점을 둔다.

GPU 와는 다른 구조를 가지며, GPU 와 cuDNN 을 사용할 때와 같은 100% 자유도를 가지고 네트워크를 구성할 수는 없다. NPU 는 딥러닝 네트워크의 각 뉴런을 하드웨어로 구현하므로, GPU 와는 다른 특성을 가지고 있다.

NPU 의 구조

행렬 연산 유닛 (Matrix Operation Unit) - **대규모 행렬 곱셈 및 누적 연산을 수행하는 유닛**으로, 뉴럴 네트워크의 핵심 연산인 Convolution 및 Fully Connected Layer 의 효율적인 처리를 담당한다. 보통 이런 연산들을 수행한다.

1. 덧셈과 뺄셈 (Addition and Subtraction) - 두 행렬의 대응하는 원소끼리 더하거나 빼준다.
2. 스칼라 곱 (Scalar Multiplication) - 모든 원소에 상수를 곱한다.
3. 행렬 곱셈 (Matrix Multiplication): 행렬 A 의 행과 행렬 B 의 열을 조합하여 새로운 행렬 C 를 만든다.

메모리 계층 (Memory Hierarchy) - 고속 캐시와 온칩 메모리 구조를 통해 데이터 접근 속도를 높인다. 데이터 이동을 최소화하여 처리 속도를 극대화한다.

데이터 흐름 아키텍처 (Data Flow Architecture) - 데이터의 흐름을 최적화하여 **병렬 처리를 극대화**한다. 각 연산 유닛이 **독립적**으로 작동하여 연산을 동시에 처리할 수 있다.

전력 관리 유닛 (Power Management Unit) - AI 연산의 전력 소모를 최소화하기 위한 다양한 전력 관리 기술이 적용된다.

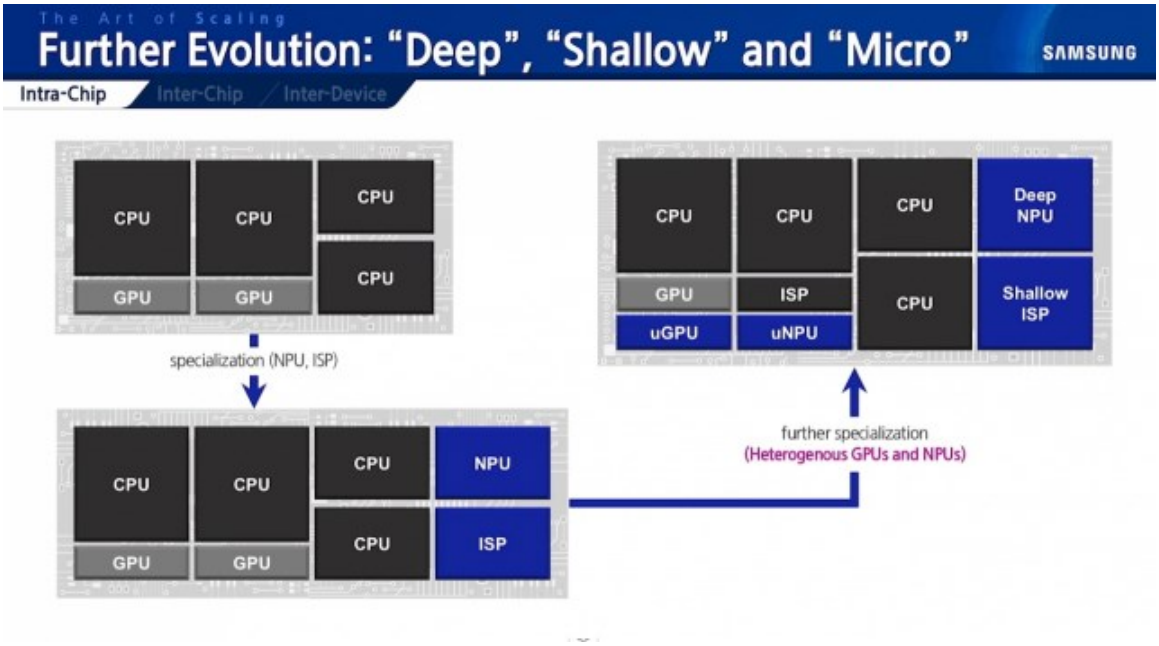
인터페이스 모듈 (Interface Module) - 다른 프로세서나 메모리와의 통신을 담당하는 인터페이스 유닛이다. PCIe, DDR, HBM 등의 다양한 인터페이스를 지원한다.

2. NPU 개발 배경 및 필요성

기존의 CPU 와 GPU 는 범용 프로세서로 설계되어 다양한 연산을 처리할 수 있는 유연성을 제공하지만, AI 연산에 최적화되지 않아 성능 및 효율성에서 한계가 있었다. AI 연산은 대규모 병렬 연산과 고속 데이터 처리 능력을 필요로 하며, 이러한 요구를 효과적으로 충족시키기 위해 NPU 가 개발되었다. NPU 는 AI 연산의 특성에 맞춘 구조로 설계되어 높은 에너지 효율성과 성능을 제공하는 전용 하드웨어이다.

칩 설계의 진화 과정

초기 칩 설계에서는 주로 CPU 와 GPU 가 사용되었으나, AI 와 이미지 처리 요구가 증가함에 따라 NPU 와 ISP 가 추가되었다. 아래 다이어그램은 삼성에서 제공한 다이어그램으로, 칩 설계의 진화 과정을 "Deep", "Shallow" 및 "Micro"라는 개념을 통해 설명한다.



초기 상태에서는 여러 개의 CPU 와 GPU 가 조합되어 있었다. 이후 NPU 와 ISP 가 추가되어 AI 연산과 이미지 처리를 전문적으로 처리하게 되었으며, 더욱 세분화된 이기종 GPU 와 NPU 를 도입하여 다양한 연산 요구를 충족하게 되었다. 이와 같은 전문화된 칩 설계는 성능과 효율성을 극대화하는 데 중요한 역할을 한다.

따라서 요점은 이렇게 정리할 수 있다. 기존의 CPU 와 GPU 는 범용 프로세서로 설계되어 다양한 연산을 처리할 수 있는 유연성을 제공하지만, AI 연산에 최적화되지 않아 성능 및 효율성에서 한계가 있다. AI 연산은 대규모 병렬 연산과 고속 데이터 처리 능력을 필요로 하며, 전력 소모를 줄이면서도 높은 성능을 제공해야 한다. 이러한 요구를 효과적으로 충족시키기 위해 NPU 는 AI 연산의 특성에 맞춘 구조로 설계되어 높은 에너지 효율성과 뛰어난 병렬 연산 능력을 제공한다. 전용 하드웨어로서 AI 연산의 특수성을 고려한 NPU 는 기존의 범용 프로세서가 제공하지 못하는 최적화된 성능과 효율성을 제공함으로써 AI 및 머신러닝 애플리케이션에서 필수적인 역할을 맡고 있다.

3. NPU 구조 분석

주요 구성 요소 및 작동 원리

행렬 연산 유닛: NPU 의 핵심 구성 요소인 행렬 연산 유닛은 AI 연산에서 가장 빈번하게 사용되는 행렬 곱셈 연산을 고속으로 처리한다. 이 유닛은 대규모 병렬 처리를 통해 대량의 행렬 연산을 동시에 수행할 수 있으며, 이를 통해 신경망의 학습 및 추론 속도를 크게 향상시킨다.

메모리 계층: NPU 는 다중 계층 메모리 아키텍처를 통해 데이터 접근 시간을 최소화한다. 이 계층은 고속 캐시 메모리와 대용량 메인 메모리를 포함하며, 연산 유닛과 메모리 간의 데이터 이동 병목 현상을 줄이기 위한 다양한 최적화 기법을 적용한다. 이를 통해 데이터 전송 지연을 줄이고 전체 시스템 성능을 향상시킨다.

데이터 흐름 아키텍처: 데이터 흐름 아키텍처는 데이터가 행렬 연산 유닛 사이를 효율적으로 이동할 수 있도록 설계되어 있다. 이 아키텍처는 데이터 경로를 최적화하여 병렬 연산을 지원하며, 데이터 이동과 연산을 동시에 수행함으로써 처리 속도를 극대화한다. 또한 데이터 흐름을 동적으로 조절하여 실시간 연산 요구사항에 대응한다.

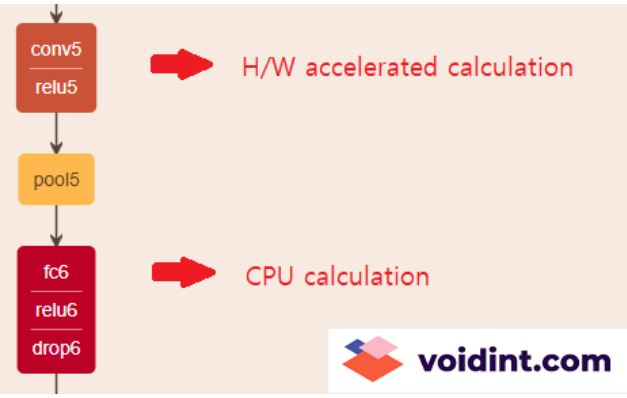
전력 관리 유닛: AI 연산은 높은 전력 소모를 동반하므로, NPU 는 다양한 전력 관리 기술을 포함한 전력 관리 유닛을 갖추고 있다. 이 유닛은 전력 소비를 실시간으로 모니터링하고,

필요에 따라 전력 소비를 최적화하기 위한 다양한 기술을 적용한다. 예를 들어 저전력 모드로의 전환, 불필요한 연산 유닛의 비활성화, 전압 및 주파수 조절 등을 통해 에너지 효율성을 극대화할 수 있다.

인터페이스 모듈: NPU 는 다양한 외부 장치와의 고속 통신을 지원하는 인터페이스 모듈을 포함하고 있다. 이 모듈은 PCIe, USB, Ethernet 등의 다양한 고속 인터페이스를 통해 다른 시스템 컴포넌트와 데이터를 교환하며 이를 통해 전체 시스템의 성능을 향상시킨다. 고속 통신 인터페이스는 NPU 의 처리 결과를 실시간으로 외부 시스템에 전달하고, 외부 데이터 소스를 신속하게 받아들일 수 있도록 설계되었다.

하드웨어 가속 계산과 CPU 계산의 구분

NPU 는 특정 연산을 하드웨어 가속 계산(H/W accelerated calculation)과 CPU 계산으로 나누어 처리한다. 아래 이미지는 딥러닝 네트워크에서 이 두 가지 방식의 구분을 시각적으로 보여준다:



이 이미지는 **딥러닝 모델 내에서 특정 연산을 수행할 때 하드웨어 가속 계산과 CPU 계산의 구분을 보여주고 있다**. 딥러닝 네트워크의 계층별로 어떤 연산이 하드웨어 가속을 통해 수행되고, 어떤 연산이 CPU 에서 수행되는지를 명확히 시각화하려는 목적을 가지고 있다. 이 그림은 딥러닝 모델의 성능 최적화를 위해 각 계층의 연산 특성에 맞는 최적의 하드웨어를 사용하는 것이 중요함을 강조하고 있다. 즉 병렬 처리가 중요한 연산은 하드웨어 가속을 통해 처리하고, 상대적으로 병렬 처리가 덜 중요한 연산은 CPU 에서 처리함으로써 전체 시스템의 효율성을 높일 수 있다는 것을 보여준다.

하드웨어 가속 계산 (H/W accelerated calculation)이란: conv5 와 relu5, 그리고 pool5 계층에서 이루어진다. 이 계층들은 대규모 병렬 처리를 요구하는 연산(ex. 컨볼루션 연산, 활성화 함수 적용, 풀링 연산)들이 주를 이루며 이를 통해 계산 효율성을 극대화할 수 있다. GPU 나 NPU 같은 특화된 하드웨어를 사용하면 이러한 연산을 매우 빠르고 효율적으로 수행할 수 있다.

CPU 계산 (CPU calculation)이란: fc6, relu6, 그리고 drop6 계층에서 이루어진다. 이 계층들은 상대적으로 덜 병렬화된 연산이나 간단한 연산(ex. 완전 연결 계층의 노드 계산, 활성화 함수 적용, 드롭아웃 적용)이 주를 이루며, CPU 에서 효율적으로 처리될 수 있다. CPU 는 다양한 연산을 유연하게 처리할 수 있는 범용 프로세서로, 특히 메모리 접근과 제어가 중요한 연산에 유리하다.

4. NPU 응용 분야

NPU 기술의 발전은 다양한 분야에서 혁신적인 응용을 가능하게 한다. 자율 주행 자동차에서는 실시간으로 대규모 데이터를 처리하여 객체 인식 및 경로 계획을 수행할 수 있다. 스마트폰에서는 이미지 및 음성 인식, 증강 현실(AR) 기능을 구현하는 데 사용될 수 있다. 또한 스마트 홈에서는 음성 인식과 이미지 분석을 통해 스마트 기기의 지능형 제어를 가능하게 하며, 의료 분야에서는 의료 이미징(CT, MRI 등)의 빠르고 정확한 진단을 돕고, 환자의 생체 신호를 실시간으로 모니터링하여 긴급 상황을 신속하게 감지하고 대응할 수 있다. 이처럼 지금까지 상상만 할 수 있던 일들을 구현해주는 중요한 역할을 수행한다.

5. NPU 와 GPU 의 장단점 비교

GPU 와 NPU 는 서로 다른 특성과 강점을 지니고 있으며, 이에 따라 다양한 응용 분야에서 선택적으로 사용된다. GPU 는 범용 프로세서로, 다양한 연산을 처리할 수 있는 유연성과 성숙한 생태계를 갖추고 있으며, CUDA 와 같은 풍부한 소프트웨어 및 도구 지원으로 인해 개발자들에게 인기가 많다. 특히 많은 코어를 활용한 높은 병렬 처리 성능으로 복잡한 그래픽 작업 및 데이터 병렬 처리에서 강력한 성능을 발휘한다. 그러나 AI 연산에 특화되지 않은 구조로 인해 전력 소비가 크고, AI 연산에서의 효율성이 떨어질 수 있다. 반면 NPU 는 AI 연산에 특화된 하드웨어로, 딥러닝과 같은 AI 작업에 최적화된 구조를 갖추고 있어 매우 높은 효율성과 성능을 제공한다. NPU 는 전력 관리 기술이 뛰어나 에너지 효율성이 높으며 AI 연산에 필요한 전용 하드웨어 요소를 통합하여 일관된 성능을 보장한다. 하지만 범용

연산에는 적합하지 않으며 상대적으로 제한된 소프트웨어 및 도구 지원으로 인해 생태계가 덜 성숙한 단점이 존재한다. 이 때문에 특정 작업에서는 GPU 와 NPU 를 조합하여 사용하는 것이 최적의 성능과 효율성을 도모할 수 있는 전략이 될 수 있다.

참고문헌

bnpassion. (2021). NPU 의 필요성과 발전 방향. Retrieved from https://m.blog.naver.com/bnpassion/222290364224

voidint. (2020). CPU, GPU, TPU, NPU 의 차이점. Retrieved from https://voidint.com/2020/10/14/cpu-gpu-tpu-npu/

voidint. (2020). GPU vs NPU: 딥러닝에서의 차이점. Retrieved from https://voidint.com/2020/11/25/gpu-vs-npu-deeplearning-difference/

ksw7713. (2023). NPU 와 GPU 의 비교. Retrieved from https://m.blog.naver.com/ksw7713/223287781137

bnpassion. (2021). NPU 의 필요성과 발전 방향. Retrieved from https://m.blog.naver.com/bnpassion/222290364224

서울경제. (2022). 삼성전자, 신형 칩 설계. Retrieved from https://www.sedaily.com/NewsView/262BEUW6TA